MULTIUSER DETECTION FOR CDMA SYSTEM

A THESIS SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF BACHELOR OF TECHNOLOGY

By

VARUN SAHAY

Roll No: 107EC006

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

2011

MULTIUSER DETECTION FOR CDMA SYSTEM

A THESIS SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF BACHELOR OF TECHNOLOGY

By

VARUN SAHAY

Roll No: 107EC006

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

2011

MULTIUSER DETECTION FOR CDMA SYSTEM

A THESIS SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF BACHELOR OF TECHNOLOGY

By

VARUN SAHAY

Roll No: 107EC006

Under the Guidance of-

Prof. POONAM SINGH

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

2011

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA 2011

CERTIFICATE

This is to certify that the thesis entitled, **"MULTIUSER DETECTION FOR CDMA SYSTEM"** submitted by **Mr. VARUN SAHAY** in partial fulfillment of the requirements of the award of Bachelor of Technology Degree in Electronics and Communication Engineering at the National Institute of Technology, Rourkela is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university or institute for the award of any Degree or Diploma.

Date: 15.5.2011

Prof. Poonam Singh

Department of Electronics and

Communication Engineering.

National Institute of Technology

Rourkela – 769008

ACKNOWLEDGEMENTS

With deep regards and profound respect, I avail this opportunity to express my deep sense of gratitude and indebtedness to Prof. Poonam Singh, Department of Electronics and Communication Engineering, N I T Rourkela for providing me the present topic and for her superb guidance, and all the invaluable suggestions. I thank her for all the constructive feedbacks provided by her during my work. Whenever I did the analysis wrong, she always remained calm and guided me to the correct path and always supported me as the best mentor. I am grateful to her.

I am grateful to Prof. S.K. Patra, Head of the Department of Electronics and Communication Engineering for allowing us to make use of the facilities available in the department to carry out the project successfully.

The staff members of the Department of Electronics and Communication Engineering were very helpful during the research work and I would like to thank them for that.

And at last but not the least, my thanks and love to all my buddies who always stood up to me and shone like the light at the end of a tunnel.

15th May 2011 Varun Sahay (107EC006)

Abstract

Multiple access interference (MAI) limits the capacity of Direct Sequence Code Division Multiple Access (DS-CDMA) systems. In CDMA systems MAI is considered as additive noise and a matched filter bank is employed. Traditionally, multiuser detectors—a code-matched and a multiuser linear filter—are used which increases the complexity of the system due to its nature of operation. Multiuser detection is an approach which uses both these filters for the optimization. However, the main drawback of the optimal multiuser detection is one of complexity so that suboptimal approaches are being sought. Much of the present research is aimed at finding an appropriate tradeoff between complexity and performance. These suboptimal techniques have linear and non-linear algorithms. In this work, we introduce Successive Interference Cancellation (SIC) which is a nonlinear suboptimal method of MUD and is based upon successively subtracting off the strongest remaining signal. Further analysis is to be carried out and simulations to be done for better understanding of SIC.

CONTENTS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1.1 What is CDMA?

CDMA stands for Code Division Multiple Access. It is a digital cellular technology that uses spread-spectrum technique means it works by digitizing multiple conversations. It is developed by Qualcomm, Inc. and standardized by the Telecommunications Industry Association (TIA) as an Interim Standard (IS-95). In this several users share the share the same physical medium i.e. same frequency band at same time.

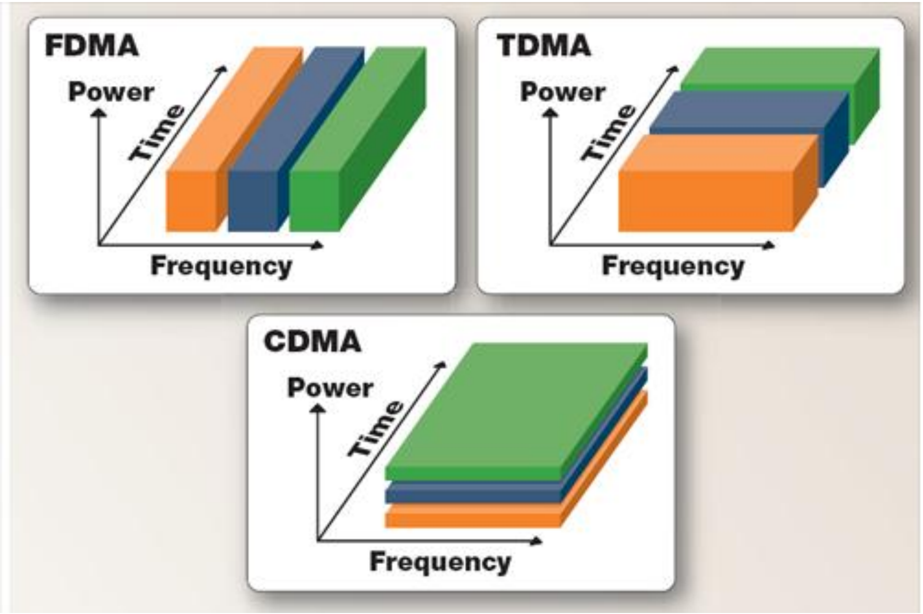

Fig. 1.1 A schematic comparison of FDMA, TDMA, and CDMA multiple-access techniques.

In CDMA every communicator will be allocated the entire spectrum all of the time. It uses codes to identify connection. A conventional DS/CDMA system treats each user separately as a signal, with other users considered as noise or MAI – multiple access interference. All users interfere with all other users and the interferences add to cause performance degradation. The near/far problem is serious and tight power control, with attendant complexity is needed to combat it. All users in a CDMA system interfere with each other. Potentially significant capacity increases and near/far resistance can theoretically be achieved if the negative effect that each user has on others can be canceled. A solution to the shortcomings of the conventional CDMA system is Multiuser Detection in which all users are considered as signals for each other.

1.2 Multiuser Detection (MUD)

Because all users are considered as signals for each other, therefore, instead of users interfering with each other, they are all being used for their mutual benefit by joint detection. The multiuser channel is just the superposition of many single user channels. (ref. [1]). Single user and multiuser spread spectrum systems have similar transmitter and receiver structures. (ref. [2]).Reduced interference leads to capacity increase of the system. It also solves the near/far problem. A cellular system has a number of mobiles which communicate with one base station (BS). The BS has to detect all the signals whereas each mobile is concerned with its own signal. This implies that the BS must know all the chip sequence. In multiuser detection, one of the main drawbacks is that of complexity. There is always a trade-off between complexity and performance of the system. Due to above mentioned two points, the main use of the multiuser detection system is for the BS, or in the reverse link (mobile to BS). The Base Station records information only on the mobiles in its own cell. This limits improvements to be expected in a MUD system.

1.2.1 MUD Concepts and Techniques:

A baseband model of a CDMA uplink is shown below. The signal received at the BS is the superposition of signals from all users, multipath components for each user's signal, and Additive White Gaussian Noise (AWGN). The figure also includes channel encoders for each transmitter.

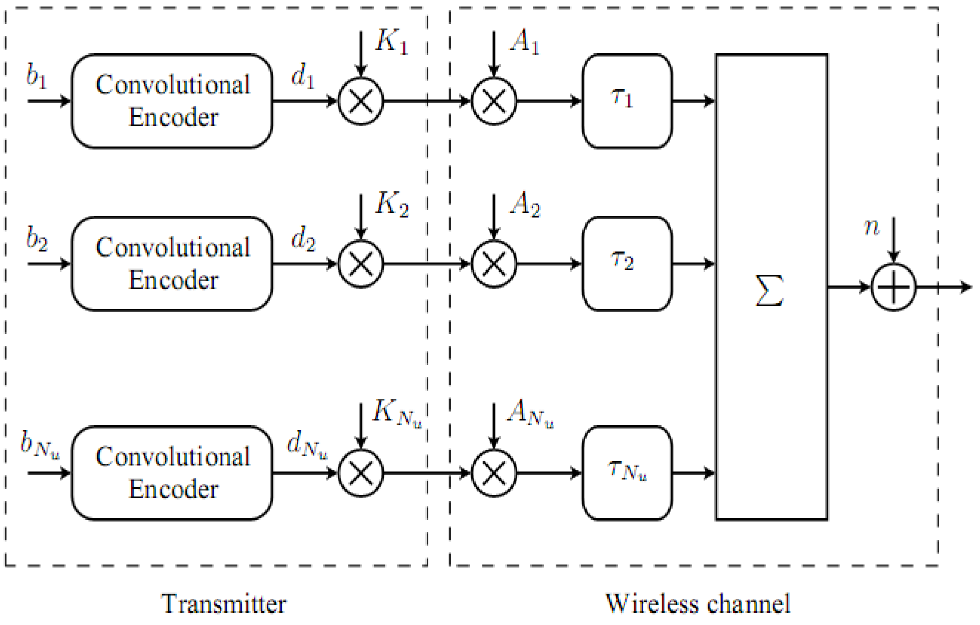

Fig 1.2 CDMA Uplink (ref. [3])

There are Nu users in the system and the data signals from these users are designated as $d_1(t), d_2(t), \ldots, d_{Nu}(t)$. The data symbols within the data signals are spread by multiplying with respective spreading sequences $K_1(t)$, $K_2(t), \ldots, K_{Nu}(t)$. The channel introduces delays $\tau_1, \tau_2, \ldots, \tau_{Nu}$ to signals from different users, and $A_1(t), A_2(t), \ldots, A_{Nu}(t)$ are the fading coefficients for the single resolvable path of each user. Spreading sequences $K_1(t)$, $K_2(t), \ldots, K_{Nu}(t)$ is given by

$$\tilde{K}_i(t) = \sum_{m=1}^{N} c_{im} p(t - (m-1)T_c)$$

Where,

c_{im} is the m^{th} chip of the spreading sequence $K_i(t)$.

N is the length of spreading sequence.

p(t) is the chip pulse shape that is assumed to be rectangular

Another BPSK model can be shown below-

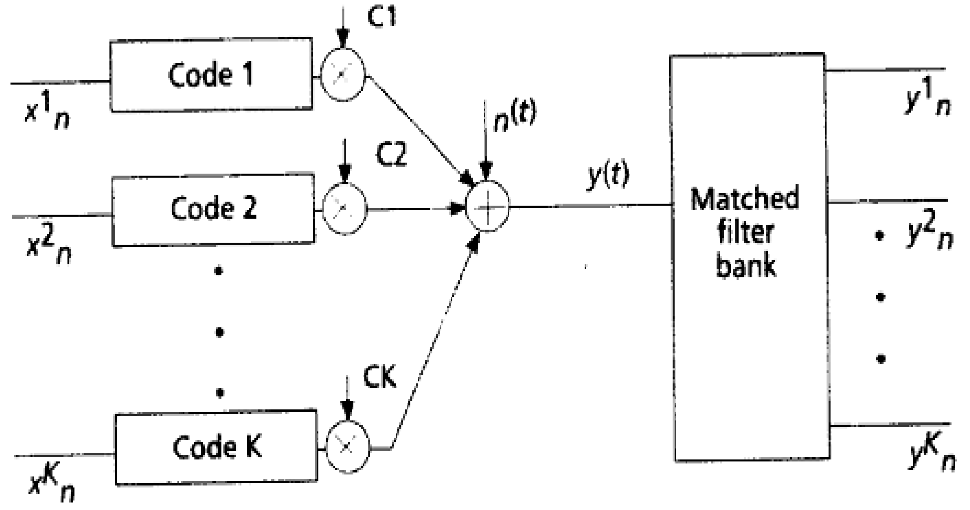

Fig 1.3 CDMA Channel Model (ref. [4])

Baseband signal for the k^th user is:

$$u_k(t)=\sum_{i=0}^{\infty}x_k(i)\cdot c_k(i)\cdot s_k(t-iT-\tau_k)$$

$x_k(i)$ is the i^th input symbol of the k^th user.

$c_k(i)$ is the real, positive channel gain.

$s_k(t)$ is the signature waveform containing the PN sequence.

τ_k is the transmission delay; for synchronous CDMA, τ_k=0 for all users.

Received signal at baseband is given by-

$$y(t)=\sum_{k=1}^{K}u_k(t)+z(t)$$

Where *K* number of users *z(t)* is the complex AWGN Sampled output of the matched filter for the k^th user:

$$y_k = \int_0^T y(t)s_k(t)dt$$

$$= c_k x_k + \sum_{j \neq k}^{K} x_j c_j \int_0^T s_k(t)s_j(t)dt + \int_0^T s_k(t)z(t)dt$$

1). 1st term - desired information

2). 2nd term - MAI

3). 3rd term - noise

Let's assume two-user case (K=2), and

$$r = \int_0^T s_1(t)s_2(t)dt$$

Outputs of the matched filters are:

$$y_1 = c_1x_1 + rc_2x_2 + z_1 \qquad y_2 = c_2x_2 + rc_1x_1 + z_2$$

Detected symbol for user k: $\hat{x}_k = \text{sgn}(y_k)$

If user 1 is much stronger than user 2 (the near/far problem), the MAI term rc_1x_1 present in the signal of user 2 is very large.

1.2.2 MUD Algorithms:

A flowchart depicting the algorithm used for MUD is given below-

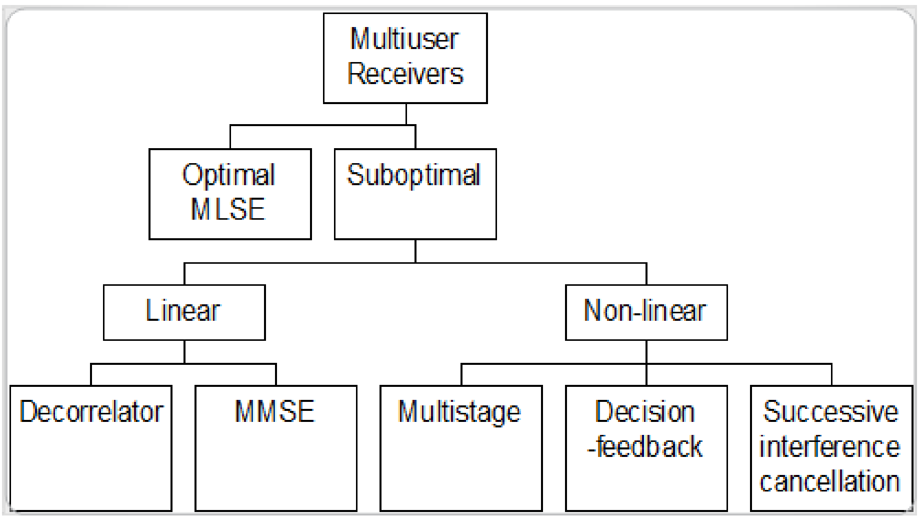

Fig 1.4 Flowchart for MUD Algorithms (ref. [5])

Our emphasis is on finding a suboptimal method to find a combination having proper complexity and performance. In this work, we mainly deal with Successive Interference Cancellation which is a nonlinear suboptimal method of MUD.

1.2.3 Optimum Multiuser Detection:

The matched filter detector, described above, was believed to be the optimum detector until proved otherwise by Verdu in the early 1980’s. His optimum solution jointly maximizes the likelihood functions for K users by choosing the bits {b1, b2, …, bK} that minimizes the mean square error (MSE) between the estimated received signal and the actual composite received signal, which is the sum of the received signals for all K users plus noise. It has been shown that the complexity of the optimum detector is O (2K), which increases exponentially with the number of users. In addition to complexity, the optimum detector requires a priori knowledge of the amplitudes of all K users, which is typically not available to the receiver. Although, the optimum detector has been shown to dramatically increase the capacity of the system, its complexity deems it infeasible to implement in the real world [3].

The work by Verdu gave hope that the capacity can ultimately increase using suboptimal multiuser detectors that balance between the two extreme cases of using the optimal detector or the matched filter detector. Hence, some linear multiuser detectors were proposed to accomplish that goal.

1.2.4 Linear Multiuser Detection:

Linear multiuser detectors attempt to attain as much of the capacity increase as the optimum detector while reducing the complexity of the system such that it can be implemented. They are simply linear filters that attempt to suppress MAI. In these detectors, a linear mapping (transformation) is applied to the soft outputs of the conventional detector to produce a better set of outputs to provide better performance. The two popular linear multiuser detectors are the decorrelating detector [4-6] and the Minimum Mean Square Error (MMSE) detector [7, 8]. They are highly analogous to the zero-forcing and MMSE equalizers used to combat inter-symbol interference (ISI) in a single- user channel [1].

The decorrelating detector attempts to completely eliminate all MAI while the MMSE detector tries to minimize the square of the residual noise plus interference. Therefore, the decorrelating detector is a special case of the MMSE detector, where the noise is zero. The decorrelating detector has the same noise enhancement problem as the zero-forcing equalizer. It is also the decorrelating detector attempts to completely eliminate all MAI while the MMSE undefined when there are more users simultaneously using the channel than spreading chip per information bit, since it is impossible to drive the interference noise to zero in this situation [3]. The MMSE, on the other hand, requires accurate channel and user information, as does the optimum detector. Along with the channel and user knowledge, the MMSE requires a KxK matrix inversion which becomes extremely complex to evaluate as K increases.

Figures 1 and 2 below show the performance comparison of the optimum detector and linear MUD to the conventional matched filter. The show the Shanon capacities as a function of the number of users K divided by the spreading factor N and the energy per bit E divided by the noise spectral density N_0, respectively.

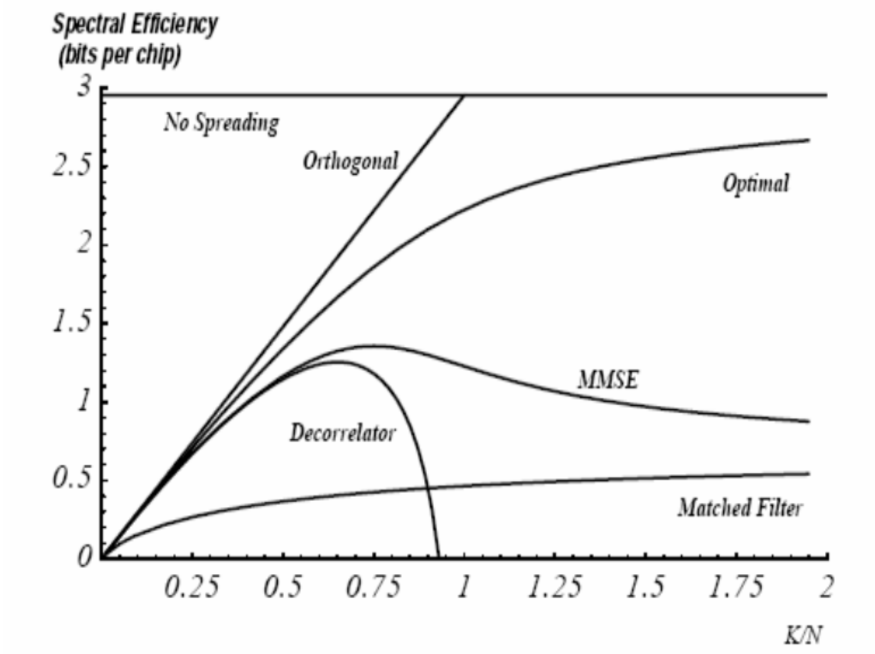

Figure 1.5 Spectral Efficiency vs. System Loading at Eb/N0 = 10 dB [3]

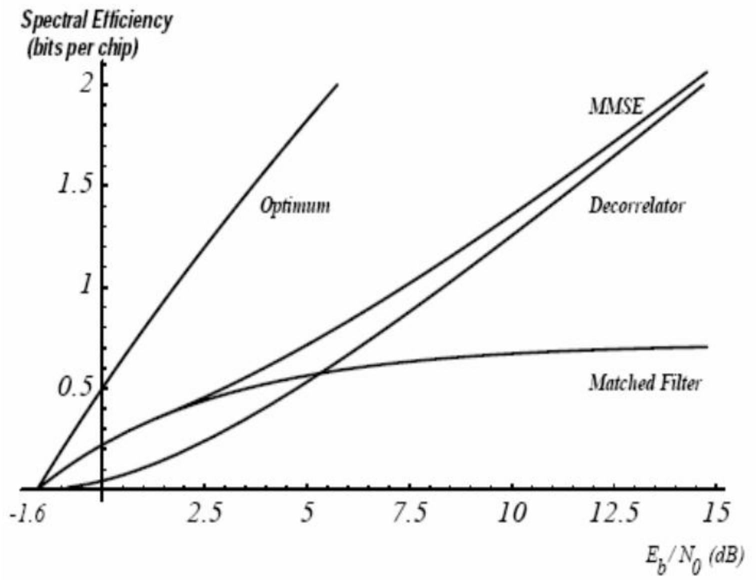

Fig 1.6 Large K Spectral Efficiencies with optimum K/N [3]

Other multiuser detection techniques include non-linear MUD, such as the Decision-Feedback (DF) multiuser detector and the turbo multiuser detector, and Interference Cancellation (IC) MUD. The next two sections will analyze the two different IC schemes, SIC and PIC, respectively in more detail. Following that will be a performance comparison of the two schemes.

1.3 Successive Interference Cancellation:

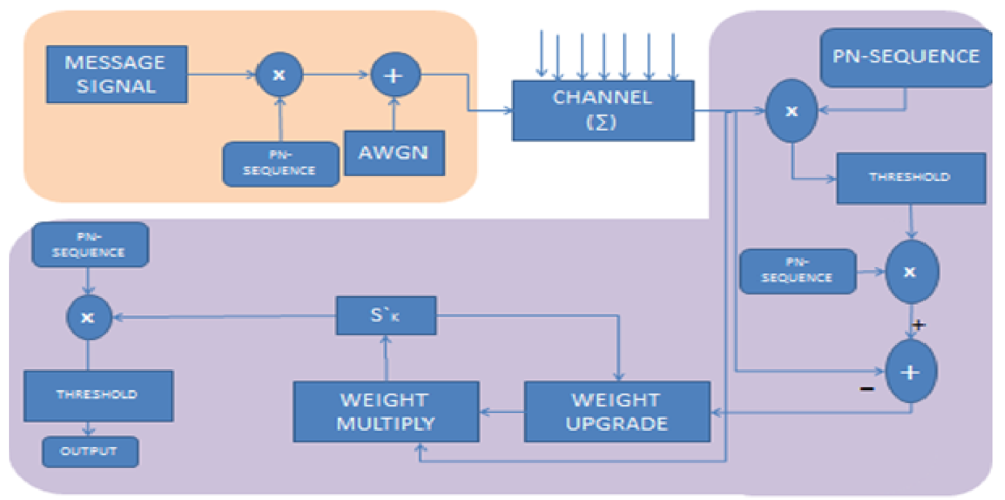

Fig. 1.7 Block diagram of Successive Interference Cancellation

Sampled output of the matched filter for the kth user:

$$y_k = \int_0^T y(t)\,s_k(t)\,dt$$
$$= c_k x_k + \sum_{j \neq k}^{K} x_j c_j \int_0^T s_k(t)\,s_j(t)\,dt + \int_0^T s_k(t)\,z(t)\,dt$$

In this equation to cancel the **Multiple Access Interference** (MAI), the factors $x_j c_j$ are needed, in addition to the cross-correlations. One of the methods could be estimating xj and cj separately. The other approach would be to estimate the product $x_j c_j$ directly by using the correlator output. The strongest signal has to be cancelled before the detection of other signals because it is most negative. The best estimate of signal strength is from the strongest signal because the best bit decision is made on that signal the strongest signal has the minimum MAI, since the strongest signal is excluded from its own MAI.

An alternative called the *Parallel Interference Cancellers* simultaneously subtract off all of the users' signals from all of the others. It works better than SIC when all of the users are received with equal strength (e.g. under power control).

Decision is made for the stronger user 1: $\hat{x}_1 = \mathrm{sgn}\left(y_1\right)$. Subtract the estimate of MAI from the signal of the weaker user:

$$\begin{aligned}\hat{x}_2 &= \mathrm{sgn}\left(y_2 - rc_1\hat{x}_1\right)\\ &= \mathrm{sgn}\left(c_2x_2 + rc_1\left(x_1 - \hat{x}_1\right) + z_2\right)\end{aligned}$$

All MAI can be subtracted from user 2 signal provided estimate is correct. MAI is reduced and near/far problem is alleviated.

2.1First Program:

The objective of this program is to compare the transmitted signal and the received signal.

2.1.1 Algorithm:

- Input data bits are taken using random data generation.
- Then the PN- sequence for the user is taken and the PN-sequence is multiplied with the data bits.
- When the input bit is one then the PN-sequence is transmitted as it is and when the data bit is zero then the reverse of the PN-sequence is transmitted i.e. it inverts.
- After this, it is assumed that the channel has AWGN so this noise is added to the signal and after this addition; the received signal corrupted with noise is obtained.
- Then threshold is applied to the received signal i.e. if the signal is greater than the threshold value it is taken as 1 and if it is less than threshold value it is taken as 0.
- Then the data bit sent by the transmitter is compared to the bits after threshold detection from this, the number of bits that are corrupted by the noise are obtained.
- Then the number of error bits are divided with the total number of output bits to get the bit error rate (BER) for the corresponding SNR (Signal-to-Noise Ratio).

2.1.2 MATLAB Program:

```
closeall;
clearall;
clc;
```

```
SNR=input('enter SNR in db');                    %signal to noise ratio in dB
input=rand(1,20);                                %taking random inputs for montecarlo simulation
between 0 and 1
signal=round(input);                             %rounding the input data to either 0 or 1
l2=length(input);                                %no of input bits
bl=ones(1,10);
ms=zeros(1,l2*10);                               %declaring message signal
pn=[1 1 0 1 0 1 1 0 1 1];                        %pn sequence of cdma user
for i=1:l2
mb=signal(i).*bl;                                %input bits extende in time by factor of length b1
mp=mb.*pn;                                       %transmitting pn sequence if signal bit is 1
otherwise zeros are transmitted
ts(1,(i-1)*10+1:i*10)=mp;                        %transmitted data
end
rs=awgn(ts,SNR);                                 %adding additive white gaussian noise to the
transmitted signal
%which acts as a noise for the channel
for i=1:l2*10
ifrs(i)<=0.5
rs(i)=0;                            %applying hard limiter to the received bits
else
rs(i)=1;                            %applying hard limiter to the received bits
end
end
```

```
for i=1:l2
sp=rs(1,(i-1)*10+1:i*10);                    %received bits contracted in time
sb=sp.*pn;                                   %decoding of data by once again multiplyin with the
user pn sequence at the reciever
rss(1,(i-1)*10+1:i*10)=sb;                   %showing grahically the received signal
end
correct=0;
for i=1:l2
ifrss(1,(i-1)*10+1:i*10)==ts(1,(i-1)*10+1:i*10)    %checking if error is there in the received
bits
correct=correct+1;
end
end
error=l2-correct                             %calculating the no. of errors
figure(1)
stem(ts);                                    %plotting TX bits
figure(2)
stem(rss);                                   %plotting RX bits
```

2.1.3 Output of the first program:

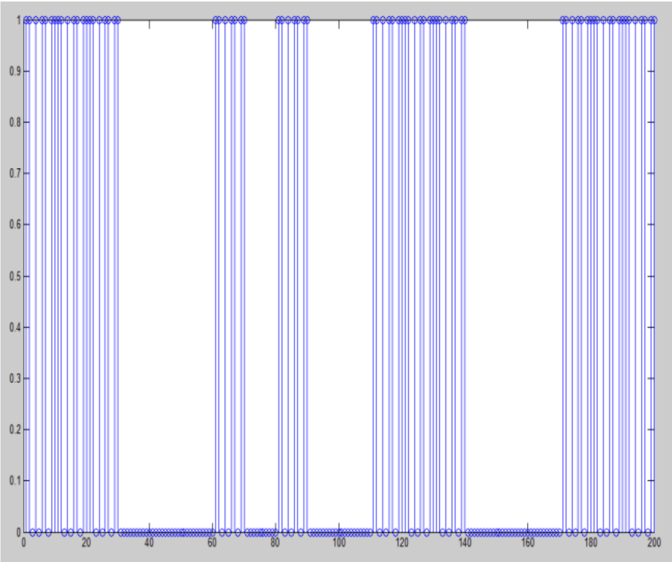

Fig2.1 (ts) (WHEN ERROR=0) i.e SNR=100

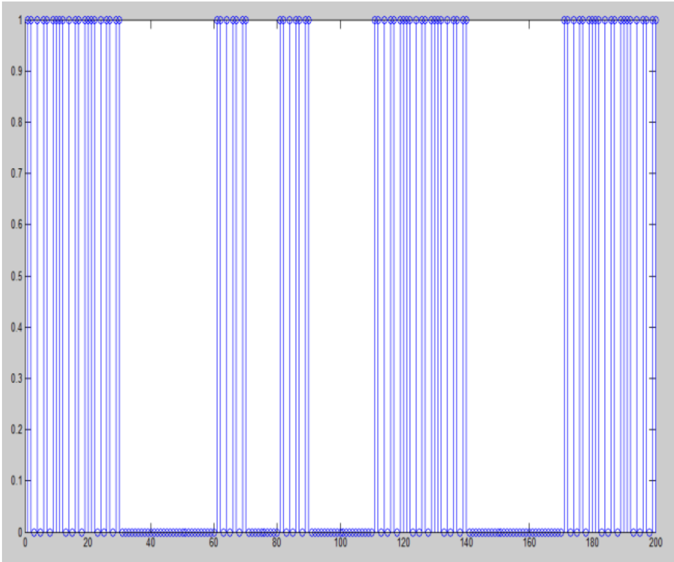

Fig2.2 (rss) (WHEN ERROR=0) i.e SNR=100

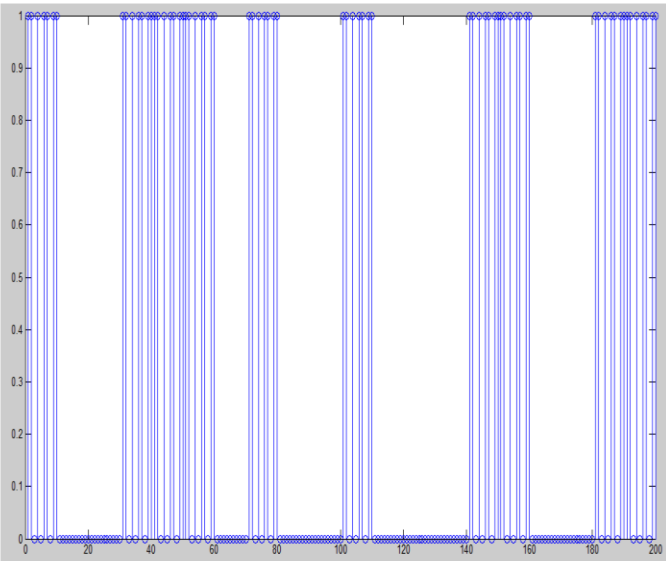

Fig2.3 (ts) (WHEN ERROR=18) i.e SNR=10

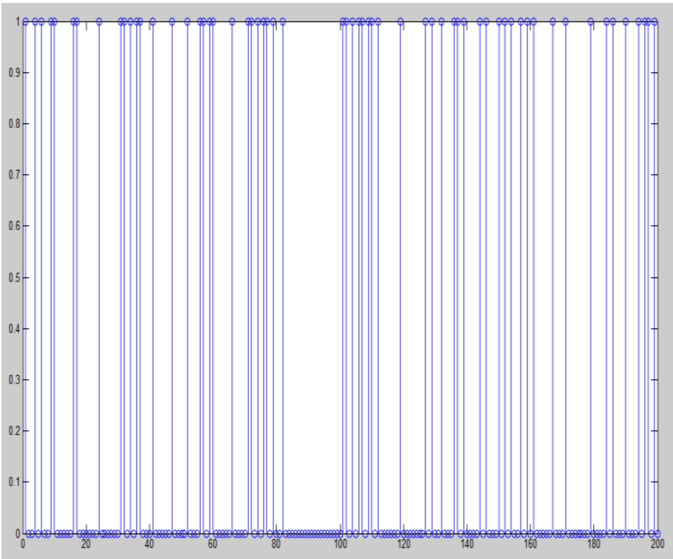

Fig2.4 (rss) (WHEN ERROR=18) i.e SNR=10

When the transmitted signal (ts) and the received signal (rss) are compared (shown in Fig 2.1 and 2.2), we get zero error because of higher SNR. So both the figures are same.

In Fig. 2.3 and 2.4, because of less SNR, 18 errors are noted as displayed above.

2.2 Second Program:

Here we plot SNR (Signal to Noise ratio) vs. BER (Bit Error Rate) for six different users.

2.2.1 Algorithm:

- Input random data are taken.
- Then the PN- sequence of the user is taken and then that is multiplied with the data bits.
- When the input bit is one then the PN-sequence is transmitted as it is and when the data bit is zero then the reverse of the PN-sequence is transmitted i.e. it inverts.
- After this, AWGN is added to the channel to corrupt signal with noise.
- Then threshold is applied to the received signal i.e. if the signal is greater than the threshold value it is taken as 1 and if it is less than threshold value it is taken as 0.
- At receiver, again the summed signal is multiplied with PN-sequence.
- Then the data bit sent by the transmitter is compared to the bits after threshold detection from this, the number of bits are obtained, that are corrupted by the noise.
- Then, the number of error bits are divided to the total number of output bits to get the bit error rate (BER) for the corresponding SNR (Signal-to-Noise Ratio).
- The graph of BER vs. SNR is plotted for six different users.

2.2.1 MATLAB Program:

```
closeall;
clearall;
clc;
SNR=[0:30];
SNR=input('enter SNR in db');                %signal to noise ratio in dB
```

```
ber_all=zeros(31,8);


for temp=1:31
    SNR=temp-11;
input=rand(1,1000);                    %taking random inputs for montecarlo
simulation between 0 and 1
signal=2.*round(input)-1;                    %rounding the input data to either 0 or 1
l1=length(SNR);
l2=length(input);                    %no of input bits
bl=ones(1,10);
ms=zeros(1,l2*10);                    %declaring message signal
U=6;
for j=1:U
    pn1=rand(1,10);
pn(j,:)=2*round(pn1)-1;                    %pn sequence of cdma user
for i=1:l2
mb=signal(i).*bl;                    %input bits extende in time by factor of length b1
tss(j,(i-1)*10+1:i*10)=mb;
mp=mb.*pn(j,:);                    %transmitting pn sequence if signal bit is 1
otherwise zeros are transmitted
tx(j,(i-1)*10+1:i*10)=mp;                    %transmitted data
end
end
```

```
rx=awgn(tx,SNR);                          %adding additive white gaussian noise to the
transmitted signal
correct=zeros(1,U);

%which acts as a noise for the channel

for j=1:U
for i=1:l2*10                          %decoding of data by once again multiplyin with the
user pn sequence at the reciever
ifrx(j,i)>=0
rx(j,i)=1;
else
rx(j,i)=-1;
end
end
end
for j=1:U
for i=1:l2
if (rx(j,(i-1)*10+1:i*10)==tx(j,(i-1)*10+1:i*10))    %checking if error is there in the recieved
bits
correct(j)=correct(j)+1;
end
end
error(j)=l2-correct(j);
end
```

```
  BER=error/l2                        %calculating the no. of errors
%figure(1)
%stem(tss(1,:));                        %plotting TX bits
%figure(2)
%stem(rx(1,:));                       %plotting
%RX bits

ber_all(temp,:)=BER;
end

for temp=1:U
figure(temp)
semilogy(-10:20,squeeze(ber_all(:,temp)),'g');
end

%
%
%    figure, semilogy(SNR(k),BER(j),'g');
%    hold on;
```

2.2.3 Output of the second program:

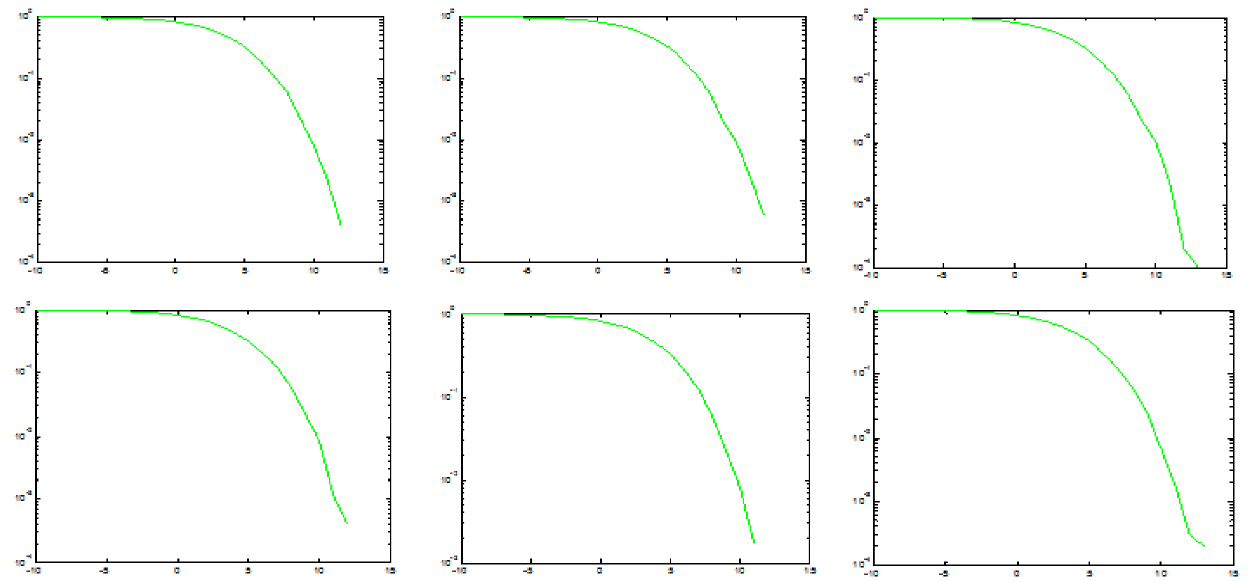

Fig. 2.5 Plot of SNR (x-axis) vs. BER (y-axis) for 6 different users

SNR(in dB)	User1	User2	User3	User4	User5	User6
1	0.9904	0.9899	0.992	0.9923	0.993	0.993
2	0.989	0.9868	0.9892	0.9884	0.9899	0.9888
3	0.9871	0.9846	0.9851	0.987	0.985	0.9841
4	0.9796	0.9818	0.98	0.9808	0.9808	0.9813
5	0.975	0.9738	0.9753	0.975	0.9745	0.9779
6	0.9656	0.9659	0.9616	0.9651	0.9686	0.9685
7	0.9534	0.9552	0.9557	0.955	0.9557	0.953
8	0.935	0.9352	0.9349	0.937	0.9355	0.9378
9	0.911	0.9064	0.9102	0.9058	0.9069	0.9042
10	0.8771	0.8762	0.8753	0.8744	0.8724	0.8766
11	0.8204	0.8202	0.8177	0.8275	0.8215	0.8164
12	0.7486	0.7495	0.7553	0.7445	0.7469	0.7501
13	0.6681	0.6744	0.6651	0.6683	0.6746	0.6676
14	0.5599	0.5602	0.5562	0.5631	0.5596	0.5556
15	0.4432	0.439	0.4398	0.4332	0.4454	0.4418
16	0.3229	0.3261	0.3212	0.3169	0.323	0.3237
17	0.2072	0.2121	0.2085	0.2029	0.2122	0.2033
18	0.1154	0.121	0.1212	0.1228	0.1246	0.1177
19	0.0641	0.0614	0.0605	0.0586	0.0589	0.0592
20	0.023	0.0216	0.0227	0.0227	0.023	0.0252
21	0.0077	0.0088	0.0099	0.0082	0.0078	0.0065
22	0.0022	0.0025	0.0022	0.0011	0.0017	0.0017
23	1.0e-003 * 0.4000	1.0e-003 *0.6000	1.0e-003 *0.2000	1.0e-003 *0.4000	0	1.0e-003 *0.3000
24	0	0	1.0e-003 *0.1000	0	0	1.0e-003 *0.2000
25	0	0	0	0	0	0
26	0	0	0	0	0	0

Table 2.1 Input from six different CDMA users and plotting the BER

2.3 Final (third) Program:

Applying the successive interference cancellation (SIC), the comparison of conventional plot of BER vs. Number of users and SIC plot of BER vs. Number of users, for different bits has been performed by MATLAB simulation.

2.3.1 Algorithm:

- A random binary signal is taken.

  Command use - rand(m,n)

  r = rand(n) returns an n-by-n matrix containing pseudorandom values drawn from the standard uniform distribution on the open interval (0,1). rand(m,n) or rand([m,n]) returns an m-by-n matrix. rand(m,n,p,...) or rand([m,n,p,...]) returns an m-by-n-by-p-by-... array. rand returns a scalar. rand(size(A)) returns an array the same size as **A**.

- Then multiply it with PN-sequence.

  Command use- seqgen.pn('Shift', 0)

Property	Description
`GenPoly`	Generator polynomial vector array of bits; must be descending order
`InitialStates`	Vector array (with length of the general polynomial order) of initial shift register values (in bits)
`CurrentStates`	Vector array (with length of the general polynomial order) of present shift register values (in bits)
`NumBitsOut`	Number of bits to output at each `generate` method invocation
`Mask` or `Shift`	A mask vector of binary 0 and 1 values is used to specify which shift register state bits are XORed to produce the resulting output bit value. Alternatively, a scalar shift value may be used to specify an equivalent shift (either a delay or advance) in the output sequence.

- AWGN noise is added.

  Command use -ys=awgn(ts(j,:),SNR(kl),lt) y = awgn(x,snr) adds white Gaussian noise to the vector signal x. The scalar SNR specifies the signal-to-noise ratio per sample, in dB. If x is complex, AWGN adds complex noise. This syntax assumes that the power of x is 0 dBW.

- In channel, the addition of different signal for different user takes place.
- At receiver, again the summed signal is multiplied with the PN-sequence.
- Now the signal is threshold to get the corresponding sender message signal.
- First the sender message signal is multiplied with the same PN-sequence.
- Then this signal is subtracted from received signal to get the error.
- Now error, the previous estimated output and the previous weight of the estimator are used for upgrading the weights.
- Then the weights are multiplied with the next received signal to get the next estimated output.
- Lastly, estimated output is multiplied with PN-sequence and after threshold, the desired output is obtained.

2.3.2 MATLAB Program:

```
closeall;
clearall;
clc;
SNR=10;
uq=1:15;

for it=1:15
    u=uq(it);
```

```
SNRL=length(SNR);


for kl=1:SNRL

kl

for i=1:u

input(i,:)=rand(1,1000);

signal1=round(input(i,:));

signal(i,:)=signal1*2-1;

end

l1=length(SNR);

l2=size(input,2);

lt=8;

bl=ones(1,lt);

ms=zeros(1,l2*lt);

w=ones(u,lt);

skn=bl;

% Construct a PN object

h = seqgen.pn('Shift', 0);


% Output 10 PN bits

set(h, 'NumBitsOut', lt);

for i=1:u

%     pn1(i,:)=(generate(h))';

pn1(i,:)=(generate(h))';

pn(i,:)=pn1(i,:)*2-1;
```

```
end

for j=1:u

ys=[];
for i=1:l2
mb=signal(j,i).*bl;
mp=mb.*pn(j,:);

ts(j,(i-1)*lt+1:i*lt)=mp;

%  ys=awgn(ts(j,:),SNR(kl),lt);
snr = 10^(0.1*SNR(kl));
nse = randn(1,length(ts(j,:)));
nse = nse - mean(nse);
No = lt/snr;
ys = ts(j,:) + sqrt(No/2)*nse;
end
rs(j,:)=ys;


end


netsig1=(sum(rs,1))/u;
```

```
for j=1:u

for i=2:l2
sp(j,(i-1)*lt+1:i*lt)=netsig1(1,(i-1)*lt+1:i*lt);
sb=sp(j,(i-1)*lt+1:i*lt)*(pn(j,:))';

ifsb<=0
sb=-1;
else
sb=1;
end
rss(j,i)=sb;
mb=rss(j,i).*bl;
mp=mb.*pn(j,:);
vs(j,(i-1)*lt+1:i*lt)=mp;

    sk(j,(i-2)*lt+1:(i-1)*lt)=sp(j,(i-2)*lt+1:(i-1)*lt)-vs(j,(i-2)*lt+1:(i-1)*lt);

w(j,(i-1)*lt+1:i*lt)= w(j,(i-2)*lt+1:(i-1)*lt)-.9*sk(j,(i-2)*lt+1:(i-1)*lt).*vs(j,(i-2)*lt+1:(i-
1)*lt);
skn(j,(i-1)*lt+1:i*lt)=w(j,(i-1)*lt+1:i*lt).*sp(j,(i-1)*lt+1:i*lt);
sb=skn(j,(i-1)*lt+1:i*lt)*(pn(j,:))';
```

```
ifsb<=0
sb=-1;
else
sb=1;
end
rss1(j,i)=sb;

end
clearvs
end




for j=1:u
correct(j)=0;
correct1(j)=0;
for i=2:l2
ifrss(j,i)==signal(j,i)
correct(j)=correct(j)+1;
end
if rss1(j,i)==signal(j,i)
correct1(j)=correct1(j)+1;
end
```

```
end
error(j)=l2-correct(j);
error1(j)=l2-correct1(j);

end
end
terr(it)=sum(error);
terr1(it)=sum(error1);

end

BER1=terr1/l2
BER=terr/l2
%figure,semilogy(SNR,error_func,'r');                %plot for the error function
%axis([-20 20 10^-6 1]);
%hold on;
figure(it),semilogy(uq,BER,'g');                     %plot for the BER using montecarlo simulation
title('plotting of error function for signal : full plot');
xlabel('number of users');
ylabel('probability of error(Bit Error Rate), Pe');
hold  on
figure(it),semilogy(uq,BER1,'r');
```

2.3.3 Output of the final program:

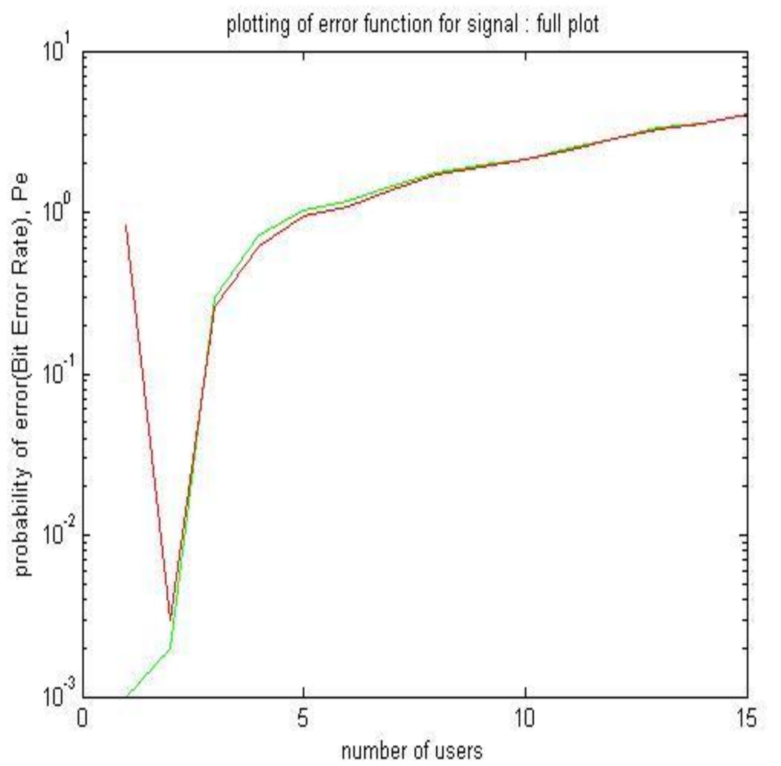

Fig. 2.6 1st Run of BER VS Number of Users

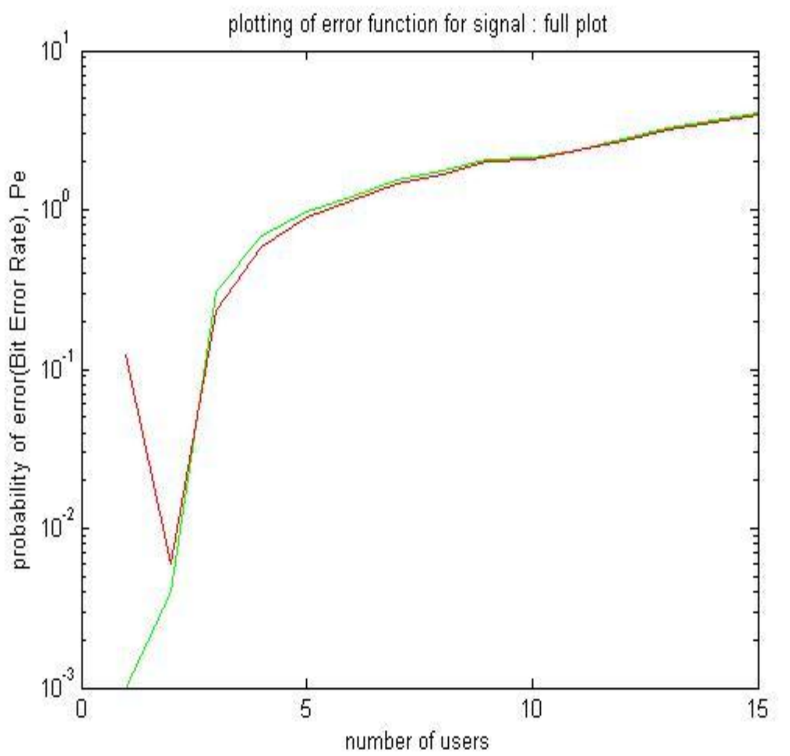

Fig. 2.7 2nd Run of BER VS Number of Users

Fig 2.6 and Fig 2.7 show the plot for BER VS Number of Users, for up to 15 users, taking 1000 information bits. The red curve indicates BER after SIC and the green curve indicates BER without SIC. In both the cases the SNR is 20 db having PN-sequence length 8 bits. A slight change is observed between the above two figures (or runs) because of random input.

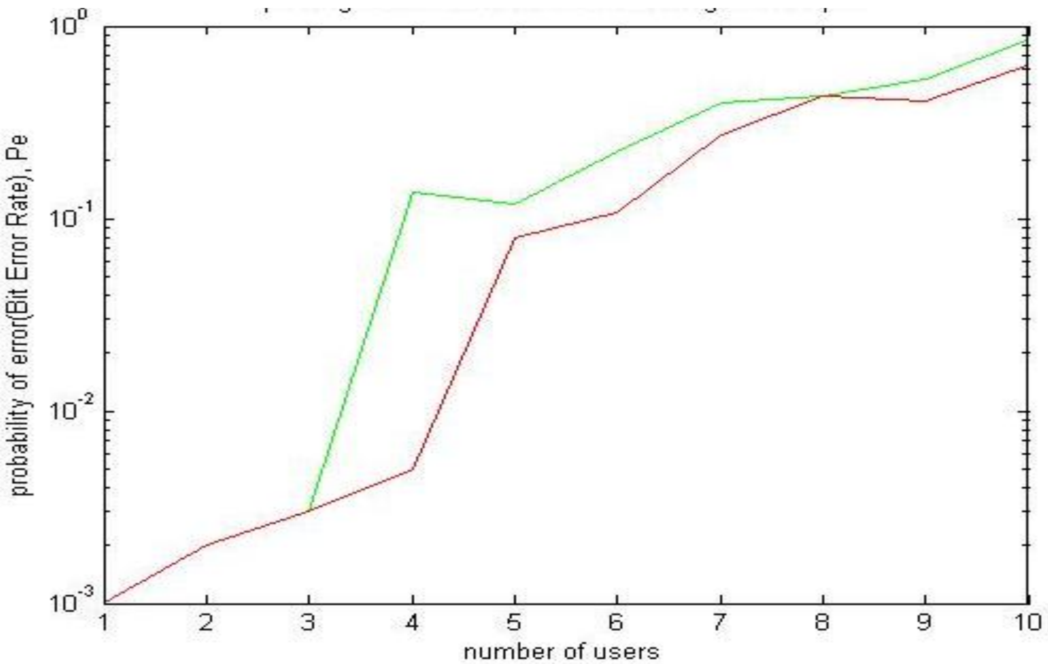

Fig. 2.8 BER VS Number of Users

Fig 2.8 gives the plot for BER VS Number of Users taking 100000 information bits, red curve indicates BER after SIC and green curve indicates BER without SIC, in both the cases the SNR is 20 db having PN-sequence length 16 bits.

It is clear from the above plot that the BER decreases when SIC technique is applied. This result shows a significant improvement in the process of Multiuser detection for CDMA systems.

Conclusions

The inclusion of SIC in a CDMA receiver can significantly improve its performance relative to that of conventional CDMA receiver where no interference cancellation is attempted. SIC appears to be more resistant to fading than PIC, and achieves better result with regards to BER and capacity performance, it suffers mightily from a high processing delay.

Future Work

While doing practical implementation, problem occurred due to processing delay, sensitivity and robustness. Capacity improvements only on the uplink would only be partly used anyway in determining overall capacity. Cost of doing multiuser detection must be as low as possible so that there is a performance/cost trade off advantage. By using better channel estimation technique the performance of the SIC can be improved further.

For delay, one of the way is to limit the number of cancellation also Group wise SIC (GSIC) has proposed to deal with delay it may degrade the performance. Investigation of GSIC and comparison it to PIC and SIC could be left for some future work.

References

[1]	Proakis J.G. Digital Communications, McGraw-Hill New York 1995
[2]	Zartash Afzal Uzmi Simplified Multiuser Detection for CDMA. Stanford University(2002)
[3]	Alexandra Duel-Hallen, Jack Holtzman and Zoran Zvonar "Multiuser Detection for CDMASystems" IEEE Personal Communications (1995-04)
[4]	Peter Ang “Multiuser Detection for CDMA Systems” IEEE Personal Communications (2001)
[5]	J. Andrews, "Successive Interference Cancellation for Uplink CDMA," Ph.D. Dissertation, Stanford University, 2002
[6]	R. Lupas, and S. Verdu, “Near-far Resistance of Multi-user Detectors in Asynchronous Channels,” IEEE J. Select Areas Commun. 38 (1990) pp. 496-508
[7]	A. Klien and B.W. Baier, “Linear Unbiased Data Estimation in Mobile Radio Systems Applying to CDMA” IEEE J. Select Areas Commun. 11(1999) pp. 1058-1066
[8]	Z. Zvonar, “Multi-user Detection in Asynchronous CDMA Frequency Selective Fading Channels” Wireless Personal Communications, Kluwer. 3 (1996) pp. 373-392
[9]	Z. Xie, et al., “A Family of Sub-optimum Detectors for Coherent Multi-user Communications,” IEEE J. Select Areas Commun. 8 (1990) pp. 683-690 .
[10]	A. Klien, et al., “Zero Forcing and Minimum Mean-Square-Error Equalization for Multi-user Detection in Code Division Multiple Access Channels”, IEEE Tran. Vehic. Tech., 45 (1996) pp. 276-287
[11]	Proakis J. G. Digital Communications 2 ed., McGraw-Hill New York 1989.